流计算模式下概率粗糙集三支决策的快速计算 *

徐健锋 [1, 2, 3]，王喜秋 [1, 3]，刘　斓 [1, 2, 3†]，汤　涛 [1, 3]

(1. 南昌大学 软件学院，南昌 330047; 2. 南昌大学 信息工程学院，南昌 330031; 3. 江西省经济犯罪侦查与防控技术协同创新中心，南昌 330031)

摘　要：在流计算模式下进行三支决策的快速计算研究是一项具有挑战性的新议题。针对流计算模式中的动态对象增量与减量同步发生的现象，提出了一种概率粗糙集三支决策的快速流计算方法。首先讨论了流计算模式中决策信息系统的单对象增减更新模式的数据模式，然后基于流计算数据变化模式分别提出了数据增量与数据减量时三支决策域的变化推理，最后基于上述理论给出了一种流计算模式下的三支决策动态增减快速学习算法。通过八种 UCI 数据集的对比实验，证明了该算法不但在时间消耗上明显优于经典三支决策算法，而且对于三支决策阈值具有较强的稳定性。

关键词：三支决策；流计算模式；动态学习；概率粗糙集

中图分类号：TP301.5　　doi: 10.3969/j.issn.1001-3695.2017.12.0855

Fast computing of probabilistic rough set three-way decision in stream computing mode

Xu Jianfeng[1, 2, 3], Wang Xiqiu[1, 3], Liu Lan[1, 2, 3†], Tang Tao[1, 3]

(1. *College of Software*, *Nanchang University*, *Nanchang* 330047, *China*; 2. *College of Information Engineering*, *Nanchang University*, *Nanchang* 330031, *China*; 3. *Jiangxi Collaborative Innovation Center for Economic Crime Investigation & Prevention & Control*, *Nanchang* 330031, *China*)

Abstract: It is a challenging topic to carry out fast computing for three-way decision in stream computing mode. Aim at the phenomenon that the increment and decrement of dynamic objects occur synchronously in the stream computing mode, this paper proposed a fast stream computing method for probabilistic rough set three-way decision. Firstly, ytdiscussed the data mode of single-object increment and decrement updating mode in stream computing. Then, proposed the reasoning of the three-way decision domains in data increment and data decrement dynamic mode respectively based on the pattern of data variation. Finally, proposed a three-way decision dynamic incremental and decremental learning algorithm based on the above theory. The comparison experiments of eight UCI datasets show that the algorithm not only outperforms the classical three-decision algorithm in time consumption, but also has strong stability for the three-way decision thresholds.

Key words: three-way decision; stream computing mode; dynamic learning; probabilistic rough set

0　引言

由加拿大贾纳大学姚一豫教授提出的三支决策[1]是在粗糙集的基础上发展出的一种不确定性问题求解的重要理论。近年来，三支决策理论在垃圾邮件过滤[2]、文本情感[3]、图像识别[4]等应用领域都取得了一系列的研究成果，这些成功的应用实例证明了三支决策在复杂背景环境中实施问题求解的重要价值。

随着大数据时代[5]的到来，新型的数据环境和计算模式不断涌现，例如流计算模式就是近年出现的一种新型动态计算形式。支持流计算模式的系统平台不断涌现和发展（如 Twitter、LinkedIn 等公司的 Storm、Kafka、YahooS4 及诞生于伯克利大学 AMPLab 的 Spark 平台等流计算平台），流计算模式的重要性愈加凸显。

流计算模式的主要动态特点可以总结为：数据源不经过外部存储器缓存，直接以滑动窗口的方式快速通过内存，而 CPU 直接对内存数据进行计算，并且实时反馈计算结果。从内存的角度观察流计算模式，可以发现流计算模式的本质是 CPU 在有限的内存空间内同时实施增量学习与减量学习（可以看做是负增量学习）的计算任务[6]，如图 1 所示。

增量学习是指一个学习系统能不断地从来自环境的新样本

收稿日期：2017-12-24；**修回日期：**2018-03-05　**基金项目：**国家自然科学基金资助项目（61763031，61673301）；江西省经济犯罪侦查与防控技术协同创新中心开放基金资助项目（JXJZXTCX-023）；江西省教育厅科技项目（GJJ161675）；江西省研究生创新专项资金项目（YC2016-S053）

作者简介：徐健锋（1973-），男，江西南昌人，副教授，博士研究生，主要研究方向为方向为粒计算、数据挖掘；王喜秋（1994-），男，江西九江人，硕士研究生，主要研究方向为机器学习、数据挖掘；刘斓（1973-），女（通信作者），江西南昌人，实验师，硕士，主要研究方向为粒计算、数据挖掘（jianfeng_X@ncu.edu.cn）；汤涛（1993-），男，安徽桐城人，硕士研究生，主要研究方向为粗糙集、粒计算、机器学习.

中学习新的知识，并能保留大部分以前已经学习到的知识，不必重新学习全部数据。降低了对时间和空间的需求,更能适应实际要求。增量学习在粗糙集及三支决策领域已经具有多年的研究历史，当前的增量学习在各类粗糙集模型[7,8]上均有相关的研究，其主要研究内容涉及上下近似[9,10]、属性约简[11,12]和决策规则[13,14]等诸多方面。但是，流计算模式的这种具有增量和减量同时实施的新型动态学习方法，尚需要进一步展开研究。所以，如何在新型的流计算模式下实施快速三支决策，是在新型计算模式下进行不确定问题求解的重要课题。

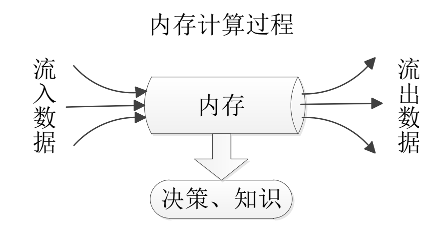


图 1　流计算模式示意图[6]

1　概率粗糙集三支决策的基本理论

概率粗糙集是构造三支决策的基础原型[15,16]。其模型基础：决策信息系统 IS 是一个四元组、$IS=(U,A,V,f)$。其中 U 代表论域中对象 x 的集合; $A=R\cup D$ 代表属性集合，其中 R 为条件属性集合，($U/R=\{R_1,R_2\cdots,R_m\}$ 为 R 属性确定的不可区分关系形成的等价类集合); D 为决策属性集合($U/D=\{D_1,D_2\cdots,D_n\}$ 为 D 属性确定的不可区分关系形成的等价类集合); V 代表 A 中各属性的取值范围; f 代表从对象到属性取值的信息函数，即 $f:U\times A\to V$。

概率粗糙集三支决策的相关定义为[1]。

定义 1　等价关系。给定信息系统 IS 上的属性子集 B，满足条件 $B\subseteq A$，则基于属性 B 的某一等价类可以表示为:

$$IND(B)=\{(x,y)\in U\times U\mid \forall a\in B, f(x,a)=f(y,a)\}$$

不同的二元关系下概率粗糙集具有不同的表达，等价关系刻画对象之间的关系。

定义 2　条件概率。给定 IS，基于条件属性 R 的任一对象集合 R_i ($R_i\in U/R$)对基于决策属性 D 的任一对象集合 D_j ($D_j\in U/D$)的条件概率定义如下:

$$P(D_j\mid R_i)=\frac{|D_j\cap R_i|}{|R_i|}$$

定义 3　三支决策域。给定一组阈值 α 和 β，其正域、边界域和负域可以分别表示为

$$POS_{(\alpha,\bullet)}(D_j)=\{x\in U\mid (x\in R_i)\wedge(P(D_j\mid R_i)\geq\alpha)\};$$
$$BND_{(\alpha,\beta)}(D_j)=\{x\in U\mid (x\in R_i)\wedge(\beta<P(D_j\mid R_i)<\alpha)\};$$
$$NEG_{(\bullet,\beta)}(D_j)=\{x\in U\mid (x\in R_i)\wedge(P(D_j\mid R_i)\leq\beta)\};$$

注：$0\leq\beta<\alpha\leq 1$

正域、边界域和负域对应的三支决策可分别解释为接收、延迟和拒绝，表示如下：

$$DES_{Accept}(R_i\to D_j),\ for\ \ R_i\subseteq POS_{(\alpha,\bullet)}(D_j),\quad (i=1,2\cdots,m;\ j=1,2\cdots,n);$$
$$DES_{Defer}(R_i\to D_j),\ for\ \ R_i\subseteq BND_{(\alpha,\beta)}(D_j),\quad (i=1,2\cdots,m;\ j=1,2\cdots,n);$$
$$DES_{Reject}(R_i\to D_j),\ for\ \ R_i\subseteq NEG_{(\bullet,\beta)}(D_j),\quad (i=1,2\cdots,m;\ j=1,2\cdots,n);$$

其中 $|U/R|=m$ 为 R 属性集所确定的等价关系商集的基数。$|U/D|=n$ 为 D 属性集所确定的等价关系商集的基数。

2　流计算模式下三支决策的增量与减量学习

2.1　决策信息系统的单对象增量与减量更新模型

流计算模式下，数据在内存计算中同时实现了数据的实时流入和实时流出。为了便于讨论，可以将流计算模式分解为对决策信息系统的执行增量更新和减量更新两个步骤的动态过程。

1)决策信息系统的单对象增量更新模型

当一个对象 x 加入到内存中的信息系统中，该新增对象记为 x_+。该信息系统在增加 x_+ 后各条件属性等价类和各决策属性等价类的变化可由下列公式更新。

$$R_i^{t+1}=\begin{cases}R_i^t\cup\{x_+\} & x_+\in R_i^t \quad 1\leq i\leq m\\ \{x_+\} & x_+\in R_i^{t+1}\quad i=m+1\end{cases}$$

$$D_j^{t+1}=\begin{cases}D_j^t\cup\{x_+\} & x_+\in D_j^t \quad 1\leq j\leq n\\ \{x_+\} & x_+\in D_j^{t+1}\quad j=n+1\end{cases}$$

其中上标 t 表示初始时刻，上标 $t+1$ 表示增加新对象后的时刻。

上述对象增量将导致条件属性等价类 R_i^{t+1} 和决策属性等价类 D_j^{t+1} 出现以下 4 种可能的数据变化情况：

情况 1　$x_+\in D_j^{t+1}\wedge x_+\in R_i^{t+1}$；

情况 2　$x_+\notin D_j^{t+1}\wedge x_+\in R_i^{t+1}$；

情况 3　$x_+\in D_j^{t+1}\wedge x_+\notin R_i^{t+1}$；

情况 4　$x_+\notin D_j^{t+1}\wedge x_+\notin R_i^{t+1}$；

注：其中 $1\leq j\leq n$ 或 $1\leq j\leq n+1$，$1\leq i\leq m$ 或 $1\leq i\leq m+1$。

性质 1　决策信息系统的单对象增量更新模型中列举的数据变化情况 3 和 4，决策规则 $R_i^{t+1}\to D_j^{t+1}$ 所属的三支决策域保持不变。

证明　上述情况 3 和 4 中由于 $x_-\notin R_i^{t+1}$ 的情况下 $R_i^{t+1}\to D_j^{t+1}$ 的条件概率 $P(D_j^{t+1}\mid R_i^{t+1})$ 保持不变，所以 $R_i^{t+1}\to D_j^{t+1}$ 的决策域也保持不变。

2)决策信息系统的单对象减量更新模型

当一个对象 x 从内存中的决策信息系统中删除后，被删除对象记为 x_-。该信息系统在删除 x_- 后各条件属性等价类和各决策属性等价类的变化可由下列公式更新。

$$\begin{cases}R_i^{t+1}=R_i^t-\{x_-\} & x_-\in R_i^t \quad 1\leq i\leq m\\ D_j^{t+1}=D_j^t-\{x_-\} & x_-\in D_j^t \quad 1\leq j\leq n\end{cases}$$

其中上标 t 表示初始时刻，上标 $t+1$ 表示删除对象后的时刻。

上述对象减量将导致条件属性等价类 R_i^{t+1} 和决策属性等价类 D_j^{t+1} 出现以下 4 种可能的数据变化情况：

情况 1　$x_-\in D_j^{t+1}\wedge x_-\in R_i^{t+1}$；

情况 2　$x_- \notin D_j^{t+1} \wedge x_- \in R_i^{t+1}$；

情况 3　$x_- \in D_j^{t+1} \wedge x_- \notin R_i^{t+1}$；

情况 4　$x_- \notin D_j^{t+1} \wedge x_- \notin R_i^{t+1}$；

注：其中$1 \le j \le n$，$1 \le i \le m$。

性质 2　决策信息系统的单对象减量更新模型中列举的数据变化情况 3 和 4，决策规则 $R_i^{t+1} \to D_j^{t+1}$ 所属三支的决策域保持不变。

2.2　三支决策的单对象增量学习策略

对于一个给定的决策等价类 D_j^t，新增一个对象 x_+，其正域、负域和边界域变化如下：

定理 1　在 *IS* 中，当 $D_j^{t+1} = D_j^t \cup \{x_+\}$ 并且 $R_i^{t+1} = R_i^t \cup \{x_+\}$ 时，

a)若 $R_i^t \subseteq POS_{(\alpha,\bullet)}(D_j^t)$，则有:

$$POS_{(\alpha,\bullet)}(D_j^{t+1}) = POS_{(\alpha,\bullet)}(D_j^t) \cup \{x_+\}$$

b)若 $R_i^t \subseteq BND_{(\alpha,\beta)}(D_j^t)$，则有：

如果 $P(D_j^{t+1} | R_i^{t+1}) \ge \alpha$，那么

$$\begin{cases} POS_{(\alpha,\bullet)}(D_j^{t+1}) = POS_{(\alpha,\bullet)}(D_j^t) \cup R_i^{t+1} \\ BND_{(\alpha,\beta)}(D_j^{t+1}) = BND_{(\alpha,\beta)}(D_j^t) - R_i^t \end{cases}$$

如果 $P(D_j^{t+1} | R_i^{t+1}) < \alpha$，那么

$$BND_{(\alpha,\beta)}(D_j^{t+1}) = BND_{(\alpha,\beta)}(D_j^t) \cup \{x_+\}$$

c)若 $R_i^t \subseteq NEG_{(\bullet,\beta)}(D_j^t)$，则有：•

如果 $P(D_j^{t+1} | R_i^{t+1}) \ge \alpha$，那么

$$\begin{cases} POS_{(\alpha,\bullet)}(D_j^{t+1}) = POS_{(\alpha,\bullet)}(D_j^t) \cup R_i^{t+1} \\ NEG_{(\bullet,\beta)}(D_j^{t+1}) = NEG_{(\bullet,\beta)}(D_j^t) - R_i^t \end{cases}$$

如果 $\beta < P(D_j^{t+1} | R_i^{t+1}) < \alpha$，那么

$$\begin{cases} BND_{(\alpha,\beta)}(D_j^{t+1}) = BND_{(\alpha,\beta)}(D_j^t) \cup R_i^{t+1} \\ NEG_{(\bullet,\beta)}(D_j^{t+1}) = NEG_{(\bullet,\beta)}(D_j^t) - R_i^t \end{cases}$$

如果 $P(D_j^{t+1} | R_i^{t+1}) \le \beta$,那么

$$NEG_{(\bullet,\beta)}(D_j^{t+1}) = NEG_{(\bullet,\beta)}(D_j^t) \cup \{x_+\}$$

证明　**a)**当 $D_j^{t+1} = D_j^t \cup \{x_+\}$ 并且 $R_i^{t+1} = R_i^t \cup \{x_+\}$ 时，根据集合的基本概念知 $|R_i^{t+1}| > |R_i^t|$ 并且 $|D_j^{t+1}| > |D_j^t|$。结合定义 2 可以得出 $P(D_j^{t+1} | R_i^{t+1}) > P(D_j^t | R_i^t)$。又因为 x_+ 的条件等价类 $R_i^t \subseteq POS_{(\alpha,\bullet)}(D_j^t)$，所以可得到 $P(D_j^t | R_i^t) \ge a$。综合上述条件可得出 $P(D_j^{t+1} | R_i^{t+1}) > P(D_j^t | R_i^t) \ge \alpha$。根据定义 4 知 x_+ 所属的决策区域为正域，所以 $POS_{(\alpha,\bullet)}(D_j^{t+1}) = POS_{(\alpha,\bullet)}(D_j^t) \cup \{x_+\}$，证毕。

b)c)的证明类似，略。

定理 2　在 *IS* 中，当 $D_j^{t+1} = D_j^t$ 并且 $R_i^{t+1} = R_i^t \cup \{x_+\}$ 时，

a)若 $R_i^t \subseteq POS_{(\alpha,\bullet)}(D_j^t)$，则有：

如果 $P(D_j^{t+1} | R_i^{t+1}) \ge \alpha$，那么

$$POS_{(\alpha,\bullet)}(D_j^{t+1}) = POS_{(\alpha,\bullet)}(D_j^t) \cup \{x_+\}$$

如果 $\beta < P(D_j^{t+1} | R_i^{t+1}) < \alpha$，那么

$$\begin{cases} BND_{(\alpha,\beta)}(D_j^{t+1}) = BND_{(\alpha,\beta)}(D_j^t) \cup R_i^{t+1} \\ POS_{(\alpha,\bullet)}(D_j^{t+1}) = POS_{(\alpha,\bullet)}(D_j^t) - R_i^t \end{cases}$$

如果 $P(D_j^{t+1} | R_i^{t+1}) \le \beta$,那么

$$\begin{cases} NEG_{(\bullet,\beta)}(D_j^{t+1}) = NEG_{(\bullet,\beta)}(D_j^t) \cup R_i^{t+1} \\ POS_{(\alpha,\bullet)}(D_j^{t+1}) = POS_{(\alpha,\bullet)}(D_j^t) - R_i^t \end{cases}$$

b)若 $R_i^t \subseteq BND_{(\alpha,\beta)}(D_j^t)$，则有：

如果 $P(D_j^{t+1} | R_i^{t+1}) > \beta$，那么

$$BND_{(\alpha,\beta)}(D_j^{t+1}) = BND_{(\alpha,\beta)}(D_j^t) \cup \{x_+\}$$

如果 $P(D_j^{t+1} | R_i^{t+1}) \le \beta$，那么

$$\begin{cases} NEG_{(\bullet,\beta)}(D_j^{t+1}) = NEG_{(\bullet,\beta)}(D_j^t) \cup R_i^{t+1} \\ BND_{(\alpha,\beta)}(D_j^{t+1}) = BND_{(\alpha,\beta)}(D_j^t) - R_i^t \end{cases}$$

c）若 $R_i^t \subseteq NEG_{(\bullet,\beta)}(D_j^t)$，则有：

$$NEG_{(\bullet,\beta)}(D_j^{t+1}) = NEG_{(\bullet,\beta)}(D_j^t) \cup \{x_+\}$$

定理 2 的证明和定理 1 的证明类似，略。

注：定理 1 对应 2.1 节第一小节中的单对象增量数据变化情况情况 1，定理 2 对应单对象数据增量变化情况 2。

当 j=n+1 或 i=$m+1$ 时，其语义为增加了新的决策等价类或者条件等价类。所以这种情况可以预设 $P(D_j^t | R_i^t)$=0，然后运用上述定理进行决策域的变换即可。

而由 2.1 节的性质 1 所述情况 3 和 4，由于结论是决策规则 $R_i^{t+1} \to D_j^{t+1}$ 所属的三支决策域保持不变，所以可以直接获得结论，不需要额外计算。

2.3　三支决策的单对象减量学习策略

对于一个给定的决策等价类 D_j^t，删除一个对象 x_-，其正域、负域和边界域变化如下：

定理 3　在 *IS* 中，当 $D_j^{t+1} = D_j^t - \{x_-\}$ 并且 $R_i^{t+1} = R_i^t - \{x_-\}$ 时，

a)若 $R_i^t \subseteq POS_{(\alpha,\bullet)}(D_j^t)$，则有：

如果 $P(D_j^{t+1} | R_i^{t+1}) \ge \alpha$，那么

$$POS_{(\alpha,\bullet)}(D_j^{t+1}) = POS_{(\alpha,\bullet)}(D_j^t) - \{x_-\}$$

如果 $\beta < P(D_j^{t+1} | R_i^{t+1}) < \alpha$，那么

$$\begin{cases} BND_{(\alpha,\beta)}(D_j^{t+1}) = BND_{(\alpha,\beta)}(D_j^t) \cup R_i^{t+1} \\ POS_{(\alpha,\bullet)}(D_j^{t+1}) = POS_{(\alpha,\bullet)}(D_j^t) - R_i^t \end{cases}$$

如果 $P(D_j^{t+1} | R_i^{t+1}) \le \beta$，那么

$$\begin{cases} NEG_{(\bullet,\beta)}(D_j^{t+1}) = NEG_{(\bullet,\beta)}(D_j^t) \cup R_i^{t+1} \\ POS_{(\alpha,\bullet)}(D_j^{t+1}) = POS_{(\alpha,\bullet)}(D_j^t) - R_i^t \end{cases}$$

b)若 $R_i^t \subseteq BND_{(\alpha,\beta)}(D_j^t)$，则有：

如果 $P(D_j^{t+1} | R_i^{t+1}) > \beta$，那么

$$BND_{(\alpha,\beta)}(D_j^{t+1}) = BND_{(\alpha,\beta)}(D_j^t) - \{x_\}$$

如果 $P(D_j^{t+1} | R_i^{t+1}) \le \beta$，那么

$$\begin{cases} NEG_{(\bullet,\beta)}(D_j^{t+1}) = NEG_{(\bullet,\beta)}(D_j^t) \cup R_i^{t+1} \\ BND_{(\alpha,\beta)}(D_j^{t+1}) = BND_{(\alpha,\beta)}(D_j^t) - R_i^t \end{cases}$$

c)若 $R_i^t \subseteq NEG_{(\bullet,\beta)}(D_j^t)$，则有：

$$NEG_{(\bullet,\beta)}(D_j^{t+1}) = NEG_{(\bullet,\beta)}(D_j^t) - \{x_-\}$$

定理 3 的证明和定理 1 的证明类似，略。

定理 4　在 *IS* 中，当 $D_j^{t+1} = D_j^t$ 并且 $R_i^{t+1} = R_i^t - \{x_-\}$ 时，

a)若 $R_i^t \subseteq POS_{(\alpha,\bullet)}(D_j^t)$，则有：

$$POS_{(\alpha,\bullet)}(D_j^{t+1}) = POS_{(\alpha,\bullet)}(D_j^t) - \{x_-\}$$

b)若 $R_i^t \subseteq BND_{(\alpha,\beta)}(D_j^t)$，则有：

如果 $P(D_j^{t+1} | R_i^{t+1}) \ge \alpha$，那么

$$\begin{cases} POS_{(\alpha,\bullet)}(D_j^{t+1}) = POS_{(\alpha,\bullet)}(D_j^t) \cup R_i^{t+1} \\ BND_{(\alpha,\beta)}(D_j^{t+1}) = BND_{(\alpha,\beta)}(D_j^t) - R_i^t \end{cases}$$

如果 $P(D_j^{t+1} | R_i^{t+1}) > \beta$，那么

$$BND_{(\alpha,\beta)}(D_j^{t+1}) = BND_{(\alpha,\beta)}(D_j^t) - \{x_-\}$$

c)若 $R_i^t \subseteq NEG_{(\bullet,\beta)}(D_j^t)$，则有：

如果 $P(D_j^{t+1} | R_i^{t+1}) \geq \alpha$，那么

$$\begin{cases} POS_{(\alpha,\bullet)}(D_j^{t+1}) = POS_{(\alpha,\bullet)}(D_j^t) \cup R_i^{t+1} \\ NEG_{(\bullet,\beta)}(D_j^{t+1}) = NEG_{(\bullet,\beta)}(D_j^t) - R_i^t \end{cases}$$

如果 $\beta < P(D_j^{t+1} | R_i^{t+1}) < \alpha$，那么

$$\begin{cases} BND_{(\alpha,\beta)}(D_j^{t+1}) = BND_{(\alpha,\beta)}(D_j^t) \cup R_i^{t+1} \\ NEG_{(\bullet,\beta)}(D_j^{t+1}) = NEG_{(\bullet,\beta)}(D_j^t) - R_i^t \end{cases}$$

如果 $P(D_j^{t+1} | R_i^{t+1}) \leq \beta$，那么

$$NEG_{(\bullet,\beta)}(D_j^{t+1}) = NEG_{(\bullet,\beta)}(D_j^t) - \{x_-\}$$

定理 4 的证明和定理 1 的证明类似，略。

注：定理 3 对应 2.1 节第二小节中的情况 1，定理 4 对应 2.1 节第二小节的情况 2。本节默认 $1 \leq i \leq m$ 且 $1 \leq j \leq n$ 。

而由 2.1 节的性质 2 所述情况 3 和情况 4，由于结论是决策规则 $R_i^{t+1} \to D_j^{t+1}$ 所属的三支决策域保持不变，所以可以直接获得结论，不需要额外计算。

3　流计算模式下三支决策动态增减学习算法

3.1　三支决策动态增减学习算法

流计算模式下，数据在内存计算中同时实现了数据的在 t 时刻后的实时流入和实时流出。借鉴时分复用的思想，将流计算模式中的一次流计算分解 t+1 时刻和 t+2 时刻二个计算步骤：即先在 t+1 时刻执行减量学习，然后在 t+2 时刻执行增量学习。根据上述思想提出以下处理流计算问题的三支决策动态增减学习算法。

算法 1：三支决策动态增减学习算法

算法输入：

t 时刻 IS 各条件等价类 U/R 及决策等价类 U/D 信息。

t 时刻每个决策等价类 D_j^t 的三支决策信息：$POS_{(\alpha,\bullet)}(D_j^t)$、$BND_{(\alpha,\beta)}(D_j^t)$、$NEG_{(\bullet,\beta)}(D_j^t)$ 及阈值 (α，β)。

$t+1$ 时刻减少的对象 x_- 及 $t+2$ 增加的对象 x_+。

算法输出：

t+2 时刻 IS 各条件等价类 U/R 及决策等价类 U/D 信息。

t+2 时刻每个决策等价类 D_j^{t+2} 的三支区域 $POS_{(\alpha,\bullet)}(D_j^{t+2})$、$BND_{(\alpha,\beta)}(D_j^{t+2})$、$NEG_{(\bullet,\beta)}(D_j^{t+2})$ 信息。

步骤 1：$t+1$ 时刻移除数据 x_- 并更新每个 $D_j^t \in U/D$ 的三支决策区域。

步骤 1.1：判断被删除对象 x_- 的条件部分属于 t+1 时刻 IS 中的哪个条件等价类。

步骤 1.2：判断被删除对象 x_- 的决策部分属于 t+1 时刻 IS 中的哪个决策等价类。

步骤 1.3：对步骤 2.1 和步骤 2.2 获得的相关条件等价类与决策等价类的每个决策规则（记为 $R_i^{t+1} \to D_j^{t+1}$）执行如下判断：

（a）如果 $t+1$ 时刻移除数据 x_- 符合 $x_- \in D_j^{t+1} \wedge x_- \notin R_i^{t+1}$ 和 $x_- \notin D_j^{t+1} \wedge x_- \notin R_i^{t+1}$，根据性质 2 则决策规则 $R_i^{t+1} \to D_j^{t+1}$ 所属的三支决策域保持不变。

（b）如果 $t+1$ 时刻移除数据 x_- 符合 $x_- \in D_j^{t+1} \wedge x_- \in R_i^{t+1}$，则根据定理 3 直接判断出决策规则 $R_i^{t+1} \to D_j^{t+1}$ 所属的三支决策域。

（c）如果 $t+1$ 时刻移除数据 x_- 符合 $x_- \notin D_j^{t+1} \wedge x_- \in R_i^{t+1}$，则根据定理 4 直接判断出决策规则 $R_i^{t+1} \to D_j^{t+1}$ 所属的三支决策域。

步骤 2：$t+2$ 时刻添加数据 x_+ 并更新每个 $D_j^{t+1} \in U/D$ 的三支决策区域。

步骤 2.1：判断添加数据 x_+ 的条件部分属于 t+2 时刻 IS 中的哪个条件等价类。

步骤 2.2：判断被添加数据 x_+ 的决策部分属于 t+2 时刻 IS 中的哪个决策等价类。

步骤 2.3：对步骤 2.1 和步骤 2.2 获得的相关条件等价类与决策等价类有关的每个决策规则（记为 $R_i^{t+2} \to D_j^{t+2}$）执行如下判断：

（a）如果 $t+2$ 时刻添加数据 x_+ 符合 $x_+ \in D_j^{t+2} \wedge x_+ \notin R_j^{t+2}$ 或 $x_+ \notin D_j^{t+2} \wedge x_+ \notin R_j^{t+2}$，根据性质 2 决策规则 $R_i^{t+2} \to D_j^{t+2}$ 所属的三支决策域保持不变。

（b）如果 $t+2$ 时刻添加数据符合 $x_+ \in D_j^{t+2} \wedge x_+ \in R_j^{t+2}$，则根据定理 1 直接判断出决策规则 $R_i^{t+2} \to D_j^{t+2}$ 所属的三支决策域。

（c）如果 $t+2$ 时刻添加数据符合 $x_+ \notin D_j^{t+2} \wedge x_+ \in R_j^{t+2}$，则根据定理 2 直接判断出决策规则 $R_i^{t+2} \to D_j^{t+2}$ 所属的三支决策域。

三支决策动态增减学习算法可以是先执行增量学习再执行减量学习，也可以先执行减量学习后执行增量学习，两者等价。本文采用的是先执行减量学习再执行增量学习的策略，其三支决策动态增减学习算法时间复杂度分析如下。

步骤 1.1 和 1.2 主要确定 x_- 对象属于哪个决策等价类和哪个条件等价类，其计算频度为 $m+n$ 。

步骤 1.3 中最好情况为子步骤（a）计算频度为 1。最坏情况为子步骤（b）和（c），其主要步骤为条件概率 $P\left(D_j^{t+1} \middle| R_i^{t+1}\right)$ 的计算，其计算频度为 $K \times \left|D_j^{t+1}\right| \times \left|R_i^{t+1}\right|$ 。

步骤 2.1 和 2.2 主要确定 x_+ 对象属于哪个决策等价类和哪个条件等价类，其计算频度为 $m+n$ 。

步骤 2.3 中最好情况为子步骤（a）计算频度为 1。最坏情况为子步骤（b）和（c），其计算频度为 $L \times \left|D_j^{t+2}\right| \times \left|R_i^{t+2}\right|$ 。

注：K 与 L 分别是为与 x_-、x_+ 相关的决策等价类和条件等价类数量，$|U/R|$=m，$|U/D| = n$ 。

为便于分析，可约定 $\left|D_j^{t+2}\right| \times \left|R_i^{t+2}\right| \approx \left|D_j^{t+1}\right| \times \left|R_i^{t+1}\right|$，所以三支决策动态增减学习算法时间复杂度为：

$$O((\mathrm{K+L}) \times (\left|D_j^{t+1}\right| \times \left|R_i^{t+1}\right|) + 2 \times (m+n))$$

3.2 三支决策经典非增量学习算法

为了便于对比讨论，本文给出流计算模式下三支决策经典非增量学习算法。即数据在内存计算 t 时刻开始实现了数据实时流入和实时流出后 $t+1$ 时刻的三支决策更新。

算法 2： 三支决策经典非增量学习算法

算法输入：

t 时刻 IS 与 $t+1$ 时刻减少的对象 x_- 及增加的对象 x_+ 及阈值 $(\alpha，\beta)$。

算法输出：

t+1 时刻 IS 各条件等价类 U/R 及决策等价类 U/D 信息。

t+1 时刻每个决策等价类 D_j^{t+1} 的三支区域 $POS_{(\alpha,\bullet)}(D_j^{t+1})$、$BND_{(\alpha,\beta)}(D_j^{t+1})$、$NEG_{(\bullet,\beta)}(D_j^{t+1})$ 信息。

步骤 1：计算 $t+1$ 时刻 IS 各条件等价类 U/R 及决策等价类 U/D 信息。

步骤 2：计算所有条件等价类与决策等价类之间的条件概率 $P\left(D_j^{t+1}\middle|R_i^{t+1}\right)$。

步骤 3：根据所有条件等价类与决策等价类之间的条件概率和阈值 (α,β) 进行匹配，完成 $t+1$ 时刻的三支区域划分。

算法 2 的算法时间复杂度分析如下。

步骤 1 计算各个等价类所需计算频度为 $2|U|$ 。

步骤 2 所有决策规则的条件概率所需最坏情况的计算频度为 $m\times n\times\left|D_j^{t+1}\right|\times\left|R_i^{t+1}\right|$ 。

步骤 3 所需计算频度为 $m\times n$ 。

注：$|U/R|=m$，$|U/D|=n$。

所以算法 2 的时间复杂度为：

$$O\Big((m\times n)\times\Big(\left|D_j^{t+1}\right|\times\left|R_i^{t+1}\right|+1\Big)+2|U|\Big)$$

约等于 $O\Big((m\times n)\times\Big(\left|D_j^{t+1}\right|\times\left|R_i^{t+1}\right|\Big)+2|U|\Big)$

与算法 1 的时间复杂度对比：

由 3.1 的分析知算法 1 的时间复杂度为 $O((\mathrm{K+L})\times(\left|D_j^{t+1}\right|\times\left|R_i^{t+1}\right|)+2\times(m+n))$，与算法 2 的时间复杂度对比显然 $(m\times n)>(K+L)$ 并且 $|U|>(m+n)$，所以算法 1 的时间复杂度明显优于算法 2 的时间复杂度。

由上述两个算法的时间复杂度分析可知：三支决策动态增减学习算法的最大的优势在于不需要对全部数据计算，而只是需要对被删除和新增的相关决策规则数据进行处理。显然三支决策动态增减学习算法计算效率的优势明显。

4 实验与分析

上述研究在理论上已经证明了三支决策动态增减学习算法能够获得经典三支决策算法相同的三支决策规则，所以本实验无须验证本算法提取决策规则的有效性是否优于经典算法，而只需要讨论其计算速度是否优于经典算法。本文提出的新算法的研究价值正是在于是否能够有效提升流计算模式下概率三支决策的计算效率。本章将使用 UCI 上的八个典型数据集进行实验来验证三支决策动态增减学习算法的有效性，以及相对于经典非增量算法在提取三支决策规则上时间花费上的优势。

操作系统为 Windows 7，机器配置为酷睿 i7-2670QM 处理器（主频为 2.2 GHz），配置的内存为 8 GB，用于实验的 Python 版本号为 3.5.2，IDE 为 spyder。

实验所使用的八个数据集来自 UCI (http://archive.ics.uci.edu/ml/datasets)。数据集 breast cancer，contraceptive method choice，mammographic mass，monk's problems，skin segmentation，thoracic surgery data，Balance Scale 和 Indian Liver Patient Dataset 的详细信息如表 1 所示。

由于上述部分数据集有些是非数值离散型数据，本文将其统一转换为等价的数值离散型数据。上述数据集中存在的少数缺失数据的情况，本文采用了众数填充的方法进行了缺失值填充处理。对于对连续型数据也进行区间离散化预处理。

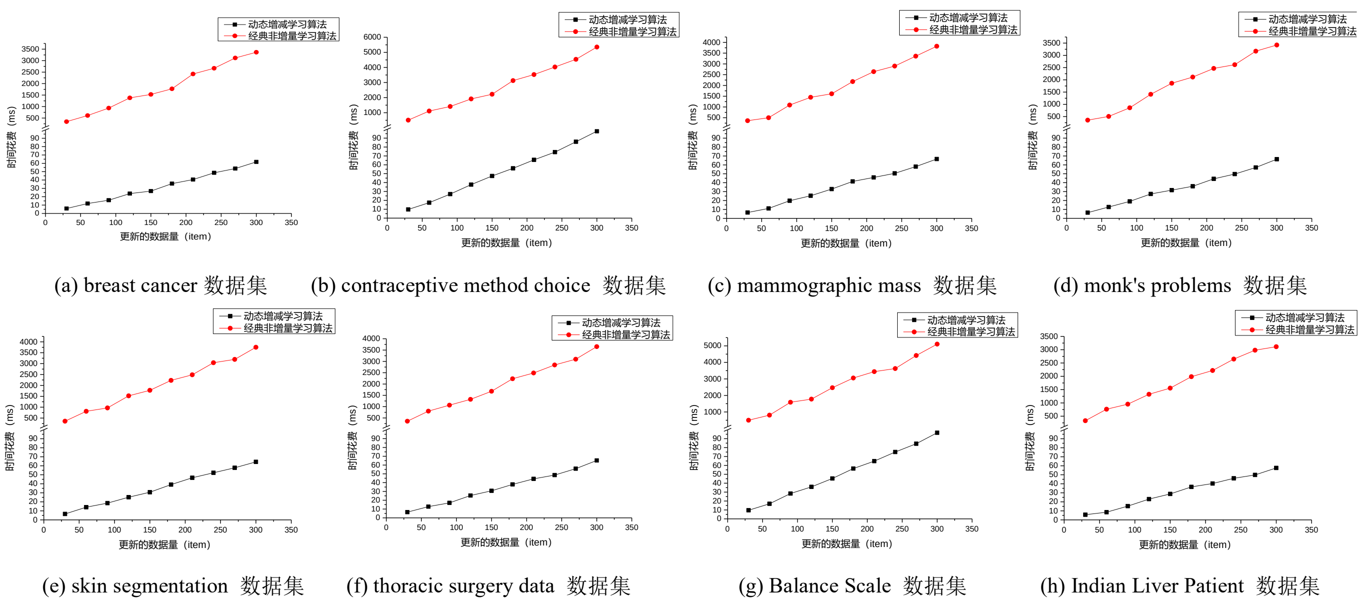
由于流计算模式的本质特点是 CPU 在有限的内存空间内同时实施增量与减量的计算任务。所以本文通过以下过程来模拟数据增量和减量动作。将内存中计算的数据量设置为固定大小。然后按照测试数据集合中各个数据对象的序列顺序，在插入新数据对象的同时删除一个内存中序号最前的数据对象。并重复上述流计算仿真过程直至数据集计算结束。

表 1 数据集信息表

编号	数据集名称	样本数量	特征数量	决策属性取值数
1	breast cancer	699	10	2
2	contraceptive Method Choice	1473	9	3
3	mammographic mass	961	6	2
4	monk's problems	432	7	2
5	skin segmentation	245057	4	2
6	thoracic surgery data	470	17	2
7	Balance Scale	625	4	3
8	Indian Liver Patient Dataset	583	10	2

4.1 三支决策动态增减学习算法与经典非增量算法对比实验

在本实验中设定内存中保存的数据量为 100 条，阈值 α 与 β 分别设置为 0.75 与 0.35。本文将收集的测试数据集以对象序号为时间的顺序，利用滑动内存窗口更新内存中的数据。三支决策动态增减学习算法及作为对比的经典三支决策学习算法，都以动态流计算模式中每次动态数据更新后完成三支决策规则提取的消耗时间作为考察指标。实验中记录点为更新数据的数量，从 0 开始，每更新 30 个数据作为一个记录点，到 300 为止共 10 组记录点。实验做 10 次取平均提取三支决策规则的耗时结果，如图 2 所示。



(a) breast cancer 数据集 (b) contraceptive method choice 数据集 (c) mammographic mass 数据集 (d) monk's problems 数据集

(e) skin segmentation 数据集 (f) thoracic surgery data 数据集 (g) Balance Scale 数据集 (h) Indian Liver Patient 数据集

图 2 动态增减学习算法与非增量学习算法时间对比

从图 2 中可以看出，两个算法的时间花费随着替换的数据量均呈现线性增长的趋势，且三支决策动态增减学习算法相对于非增量学习算法的时间花费有大幅度的降低。

由算法的时间复杂度可知，随着内存中的数据更新，三支决策经典非增量学习算法需要进行所有数据等价类的重新划分及条件概率的重新计算，此过程每次执行时间复杂度为 $O\left(\left(m\times n\right)\times\left(\left|D_j^{t+1}\right|\times\left|R_i^{t+1}\right|\right)+2\left|U\right|\right)$，其时间消耗较大，且与内存中的数据量及等价类的划分相关；而三支决策动态增减学习算法由于只需要对当前实时变化的数据对象进行决策域的更新，最多只需执行时间复杂度为 $O((K+L)\times(|D_j^{t+1}|\times|R_i^{t+1}|)+2\times(m+n))$，节省了更新时间。

4.2 不同阈值下的三支决策动态增减学习算法平均时间花费实验

为验证不同阈值对三支决策动态增减学习算法时间效率的影响，(α,β) 将分别采取如下 5 组取值 $\{(0.6,0.4),(0.7,0.3),(0.8,0.2),(0.9,0.1),(1,0)\}$ 进行实验，内存中数据量与实验 1 一致，仍定为 100 条。实验将计算更新 300 条数据下三支决策动态增减学习算法的时间花费，每组做 10 次，取实验结果的均值进行对比，以此判断阈值对三支决策动态增减学习算法时间复杂度的影响。实验结果如图 3 所示，其中 A-H 分别为表一中的数据集 1-8。

根据 3.2 小节的时间复杂度分析可知，时间复杂度和内存中数据集的大小等有关，与阈值并无关联。从图 3 可以看出，在不同的 5 组阈值下，基于给定的 8 组数据集，更新同样大小数据的时间花费基本没有太大的变化，与理论分析相符。

另外通过本实验结果也显示在不同阈值下三支决策动态增减学习算法的时间消耗差异不大，也证明了本算法的对于三支决策阈值的设定具有较好的鲁棒性和适用性。

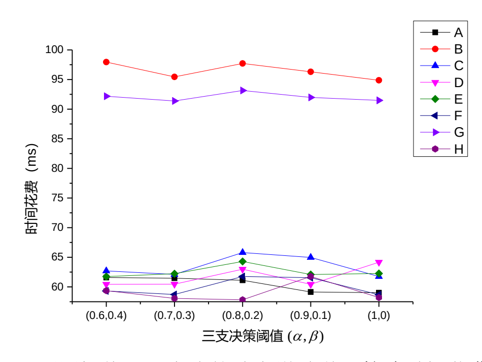


图 3 不同阈值下三支决策动态增减学习算法时间花费

5 结束语

本文中以概率粗糙集决策信息系统模型为基础，以三支决策单对象增量、减量的流计算模式为研究对象，实施了流计算模式下的三支决策区域变换决策的推理。并且提出了一种流计算模式下快速三支决策的动态增减学习算法。通过与经典三支决策方法的理论与实验对比，证明了本文提出的三支决策动态增减学习算法不但能够获取的等效的三支决策，而且能够流计算模式下极大的提高计算的时间效率。

随着流计算平台的发展，流计算模式在机器学习和大数据分析领域的应用将越来越得到广泛重视。作为不确定问题求解的重要理论，流计算模式中进行三支决策理论研究不但给流计算模式平台提出了一种不确定问题快速求解的新方法，而且丰富了三支决策的理论体系。

参考文献：

[1] Yao Yiyu. An outline of a theory of three-way decisions [M]// Rough Sets and Current Trends in Computing. Berlin: Springer, 2012: 1-17.

[2] Zhou Bing, Yao Yiyu, Luo Jigang. Cost-sensitive three-way email spam filtering [J]. Journal of Intelligent Information Systems, 2014, 42 (1): 19-45.

[3] Zhang Zhifei, Miao Duoqian, Nie Jianyun, *et al.* Sentiment uncertainty measure and classification of negative sentences [J]. Journal of Computer Research & Development, 2015, 52 (8): 1806-1816.

[4] Li Huaxiong, Zhang Libo, Huang Bing, *et al.* Sequential three-way decision and granulation for cost-sensitive face recognition [J]. Knowledge-Based Systems, 2016, 91 (C): 241-251.

[5] Assunção M D, Calheiros R N, Bianchi S, *et al.* Big data computing and clouds: trends and future directions [J]. Journal of Parallel & Distributed Computing, 2015, 79–80: 3-15.

[6] 孙大为, 张广艳, 郑纬民. 大数据流式计算: 关键技术及系统实例 [J]. 软件学报, 2014, 25 (4): 839-862.

[7] Zhang Junbo, Zhu Yun, Pan Yi, *et al.* Efficient parallel boolean matrix based algorithms for computing composite rough set approximations ☆ [J]. Information Sciences, 2016, 329: 287-302.

[8] Li Shaoyong, Li Tianrui, Hu Jie. Update of approximations in composite information systems [J]. Knowledge-Based Systems, 2015, 83 (1): 138-148.

[9] Zeng Anping, Li Tianrui, Hu Jie, *et al.* Incremental updating fuzzy rough approximations for dynamic hybrid data under the variation of attribute values [J]. Information Sciences, 2016, 378 (C): 363-388.

[10] Luo Chuan, Li Tianrui, Chen Hongmei, *et al.* Efficient updating of probabilistic approximations with incremental objects [J]. Knowledge-Based Systems, 2016, 109: 71-83.

[11] Chen Hongmei, Li Tianrui Luo, Chuan, *et al.* A decision-theoretic rough set approach for dynamic data mining [J]. IEEE Trans on Fuzzy Systems, 2015, 23 (6): 1958-1970.

[12] Li Meizheng, Wang Guoyin. Approximate concept construction with three-way decisions and attribute reduction in incomplete contexts [J]. Knowledge-Based Systems, 2016, 91: 165-178.

[13] Das R T, Kai K A, Chai Q. ieRSPOP: a novel incremental rough set-based pseudo outer-product with ensemble learning [J]. Applied Soft Computing, 2016, 46: 170-186.

[14] Azam N, Zhang Yan, Yao JingTao. Evaluation functions and decision conditions of three-way decisions with game-theoretic rough sets [J]. European Journal of Operational Research, 2017, 261 (2): 704-714.

[15] Yao Yiyu. Probabilistic rough set approximations [J]. International Journal of Approximate Reasoning, 2008, 49 (2): 255-271.

[16] Yao Yiyu. The superiority of three-way decisions in probabilistic rough set models [J]. Information Sciences, 2011, 181 (6): 1080-1096.